OLFAR: ADAPTIVE TOPOLOGY FOR SATELLITE SWARMS

A. Budianu
University of Twente,
Faculty of Electrical Engineering, Mathematics and Computer Science,
Telecommunication Engineering Group,
P.O. Box 217, 7500 AE, Enschede, The Netherlands
E-mail: a.budianu@utwente.nl

R. T. Rajan[1], S. Engelen[2], A. Meijerink[3], C. J. M. Verhoeven[2] and M. J. Bentum[4]

***Abstract*—Low-frequency space observation is one of the developing directions in radio astronomy, as scientists try to reveal more of the universe. The Orbiting Low Frequency Array for Radio Astronomy project is aimed at developing a distributed radio telescope sensitive to ultra-long waves, by placing an array of antennas, far away from any terrestrial interference. It will consist of over 50 small satellites grouped in a swarm capable of collecting and processing astronomical data. The system will use its resources to the limit and will need an efficient communication topology in order to guarantee functionality. In this paper we present a clustering scheme that reduces data distribution efforts at the cost of decreasing imaging redundancy.**

I. Introduction

The Ultra Low Frequency band below 30 MHz is one of the last unexplored frequency band in radio astronomy, due to ionospheric distortion, man-made interference and even solar flares. An unequivocal solution to the problem is to place antennas far away from Earth to observe at these long wavelengths. The Orbiting Low Frequency Antennas for Radio astronomy (OLFAR) [1] project is aimed at designing and developing a detailed system concept for an array of scalable autonomous nano satellites in space (not more than 100 km apart), to be used as a scientific instrument for ultra-low-frequency observations. The OLFAR swarm could either orbit the moon, whilst sampling during the Earth-radio eclipse phase, or orbit the sun, Earth-trailing or -leading, sampling almost continuously. To avoid single point of failure, OLFAR will be a distributed system with satellites cooperating with each other, both for radio astronomy observations and communications. The swarm will employ distributed correlation [2], with inter-satellite data rates in excess of 6 Mbits/sec for an instantaneous observation bandwidth of larger than 1 MHz and a data resolution higher than 1 bit. At the system level, each nano satellite [3] is by definition power constrained. In addition, the satellites are mobile and thus the topographical distribution keeps changing with time. All these requirements make data distribution within the swarm very complex.

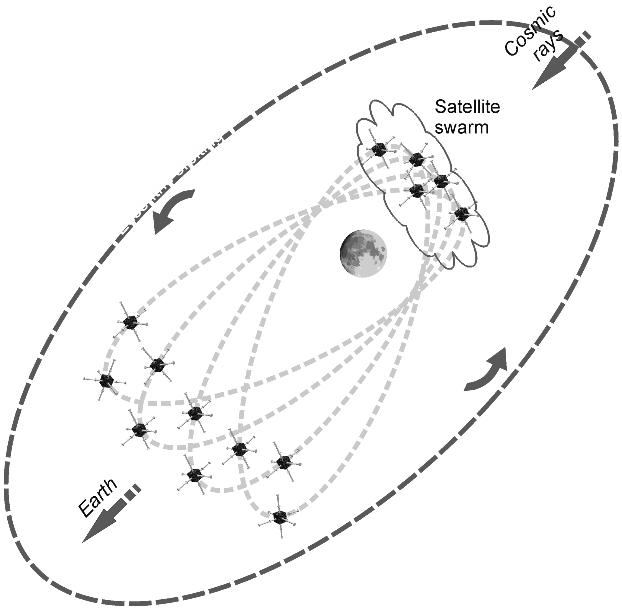

Fig. 1. Satellite swarm orbiting around the Moon. Eccentric orbiting causes the satellite cloud to vary its distribution with position.

As already described, the OLFAR project will consist of a large number (50 to 1,000) [4] of small satellites, that will gather data individually and process it in a collective manner, ensuring this way the functionality of a low-frequency radio telescope. On the whole, the system will act as a large wireless sensor network, with a few peculiar requirements.

[1] R. T. Rajan is with Delft University of Technology, Kluyverweg 1, 2629 HS Delft, The Netherlands and ASTRON, P.O. Box 2, 7900 AA Dwingeloo, The Netherlands (e-mail: rajan@astron.nl).

[2] S. Engelen and C. J. M. Verhoeven are with the Delft University of Technology, Kluyverweg 1, 2629 HS Delft, The Netherlands (e-mail: s.engelen@tudelft.nl; c.j.m.verhoeven@et.tudelft.nl).

[3] A. Meijerink is with the Telecommunication Engineering Group, Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, 7500 AE Enschede, The Netherlands (e-mail: a.meijerink@utwente.nl).

[4] M. J. Bentum is with the Telecommunication Engineering Group, Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, 7500 AE Enschede, The Netherlands and with ASTRON, P.O. Box 2, 7900 AA Dwingeloo, The Netherlands (e-mail: m.j.bentum@utwente.nl).

First of all, the network of satellites will be positioned in space, orbiting either the sun or the moon [3]. Little is known about the environment from the communication point of view, and, thus, the behavior of the satellites will have a high degree of uncertainty. Added to this, offline debugging and component replacement is practically impossible. Hence, the system designer has to consider the harshest conditions that may appear.

Secondly, the distances between the nodes of the network will be up to 100 km. This is imposed by the low-frequency telescope functionality. In order to achieve sufficient spatial resolution, the telescope needs to have an aperture diameter of 10–100 km [1]. All the satellites will then be distributed over a very large cloud, making the network a low-density one. This also adds complexity to the communication problem.

Finally, space observation in low frequencies requires gathering and processing large amounts of data. What is more, the satellites need to transmit their observed astronomical data to all the other members of the swarm. The necessary throughput of every node will be very high. The observed data D_{obs} rate will exceed 6 Mbit/second/satellite [2], while the required inter-satellite reception rate D_{in} will be

$$D_{\mathrm{in}} = \frac{N_{\mathrm{sat}} - 1}{N_{\mathrm{sat}}} D_{\mathrm{obs}} \,, \tag{1}$$

where N_{sat} is the number of satellites.

Apart from the up-mentioned requirements, the swarm will also have constraints that most sensor networks encounter. The power of each satellite will be limited and should be used mainly to serve the purpose of the system—space observation. Mobility is another characteristic that has to be taken into consideration. Nodes will drift away from the ideal trajectory and will move relative to each other [5]. The topographical distribution of the satellites will not be constant in time.

All these requirements make the communication task very difficult to fulfill.

The rest of this paper is organized as follows: in Sections II, III and IV, we motivate our clustering approach. In Section V, a new dynamic clustering algorithm is described. Simulation results are presented in Section VI, and concluding remarks are drawn out in Section VII.

II. Problem formulation

The most straightforward way to solve the communication layer problem would be to employ a full-mesh topology for the network. This copes well with the fact that all the satellites are identical and also provides robustness to the system. However, it comes with high demands of communication resources. It is certainly not feasible for a large network distributed over a vast area in space, because the necessary power will tend to increase exponentially with the number of nodes and the distances between them.

One may suggest that a Gossip protocol would be a better solution [6]. It is successfully used in some ad-hoc networks that implement distributed algorithms. Unfortunately, the need for low latency and high data rates makes it an inappropriate solution for the OLFAR network, at least for the permanent regime. Although it cannot guarantee the correct functioning of the swarm as a radio telescope, gossiping remains a good solution for the initialization of the system—network discovery and synchronization.

For large-scale mobile ad-hoc networks (MANETs), a clustering approach is essential to guarantee basic levels of system performance, such as throughput or delay. In most cases, network division offers important benefits. A cluster structure eases spatial reuse of resources and can increase the system's capacity. Same frequencies or codes can be deployed in disjoint clusters. Added to this, routing is more facile to do in a hierarchical approach. Cluster heads or gateways can form a back-bone for the system, simplifying the data distribution task. Nonetheless, one of the most important advantages of clustering is that it makes the network more stable and smaller from the members' point of view. Local changes in one domain will not cause disturbances in the entire system [7].

Therefore, clustering might be a good solution for many WSNs, but, in order to be suitable for the OLFAR satellite swarm, it should also be power-efficient. Our goal is to reduce the complexity of the communication task and minimize the energy consumed for transmitting and receiving information. In this way, the system's resources will be used more efficiently to achieve the science objective, namely, exploring the universe in the low-frequency band.

III. Model

To prove the power efficiency of a clustering scheme, the following test scenario has been considered: a random sensor network with N nodes was generated. A full-mesh topology and a clustered topology were employed to the system and then the total necessary power for communication was calculated. By necessary power we refer to the sum of power needed for transmitting the data to all members of the network and the power needed for receiving and processing the data.

The following assumptions have been made:

1) The network terminals are uniformly distributed on a L by L square surface.
2) The communication environment is contention-free and error-free, hence, there is no need for data retransmission.
3) In the full-mesh topology, each sensor transmits and receives data from all the other sensors
4) Clusters are attained by dividing the initial surface into M equal squares, M taking only the values 4 and 16. This is illustrated in Fig. 2. As the clusters are formed considering a geometric criterion, the number of nodes may vary from cluster to cluster.
5) For each cluster, the the node closest to the center of gravity $\mathbf{r}_{\mathrm{G}_k}$ of the cluster is elected as the cluster head, where:

$$\mathbf{r}_{\mathrm{G}_k} = \frac{1}{N_k} \sum_{i=1}^{N_k} \mathbf{r}_i \,. \tag{2}$$

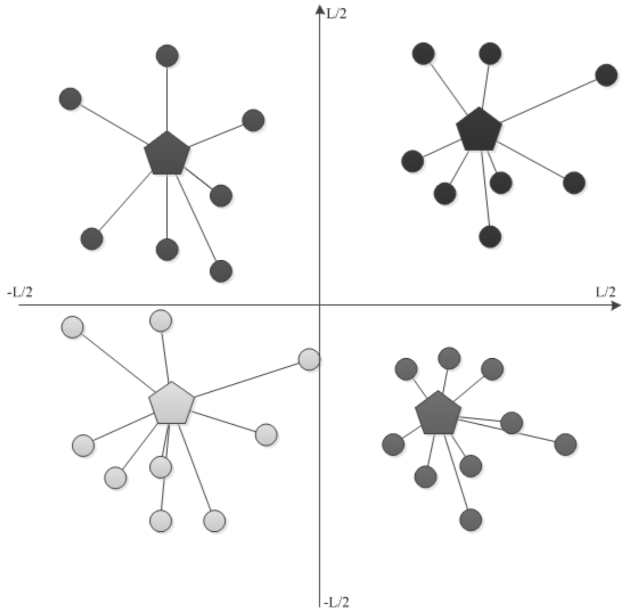

Fig. 2. Clustering example. The area over which the sensor network is spread was divided into four equal subareas, each one corresponding to a different cluster. The master nodes are represented by pentagon shapes, whereas the slaves are represented by circles.

Index k denotes a particular cluster, N_k is the number of nodes in cluster k, and $\mathbf{r}_i$ corresponds to node i in cluster k.

6) A node that is not a cluster head is considered to be a slave node. All slave nodes send and receive data only to and from their corresponding cluster head. In other words, each cluster employs a star topology.
7) Each cluster head sends and receives data to all slaves in its cluster and to all the other cluster heads.
8) Each sensor consumes energy E for receiving and processing one unit of data.
9) Each sensor has a minimum transmission power that corresponds to a transmission distance $d_{\min}$. If the distance d_{ij} between the sending sensor i and receiving sensor j is smaller than $d_{\min}$, energy E is consumed for transmitting one unit of data. Otherwise, the consumed energy for transmitting one unit of data increases quadratically, by the following rule:

$$E_{\text{TX}} = \left(\frac{d_{ij}}{d_{\min}} \right)^2 E \tag{3}$$

IV. RESULTS

Considering all the assumptions, we calculate the total necessary power for communication duty for both cases: full-mesh topology and clustered network. In Fig. 3, the ratio between the two is plotted as a function of $d_{\min}/L$. The value is higher than one proving that a clustered network is more power efficient. For low values of $d_{\min}/L$ (low-density networks), the required power for a full-mesh topology tends to be one magnitude order higher than for the clustered case.

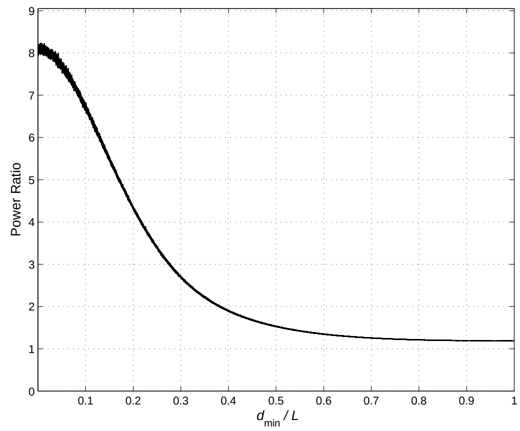

Fig. 3. Power-efficiency of clustering: the ratio between the necessary power for a full-mesh network and the necessary power for a clustered network as a function of $d_{\min}/L$.

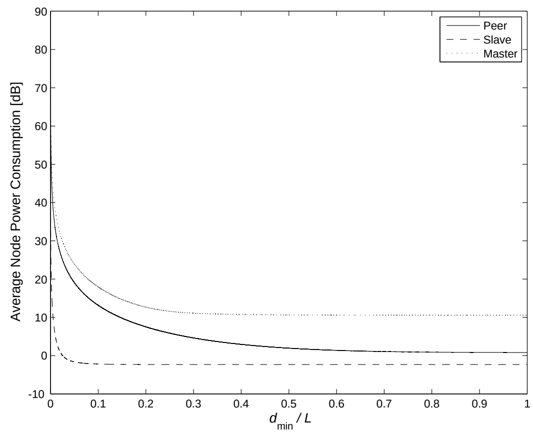

Fig. 4. Power requirements: average node power consumption as a function of $d_{\min}/L$.

For the scenario described above, we also calculated the average power consumed by every type of node: slave and master for clustered network, and peer node for the full-mesh case. In Fig. 4, the values are plotted on a logarithmic scale, as a function of $d_{\min}/L$. Clustering a network will make most of the members consume less power for communication, at the cost of overloading the relay nodes. There is a trade-off, yet, for low values of $d_{\min}/L$ the advantage is clearly in favor of clustering.

The average power consumed by a slave node is very low compared to a peer node or a master node. Thus, in a clustered network the slaves could save their resources for fulfilling other tasks than data distribution. Master nodes, on the other hand, will tend to consume their energy mostly for communication duty. For that reason, masters will not contribute to the observation task.

Despite the basic clustering scheme we used, the results show that dividing a network into multiple clusters increases its efficiency.

V. A DYNAMIC CLUSTERING SCHEME FOR THE SATELLITE SWARM

Clustering is a step forward in achieving a functional distributed radio telescope in space. By dividing the OLFAR swarm into small groups of satellites, and electing certain gateway satellites to route the astronomical data, energy resources of the system will be used more efficiently. This hierarchical approach, though an effective tool for the communication layer, comes with one major drawback. Some of the satellites that will act as group leaders, will not be able to actively participate in the scientific task of the swarm. However, as long as this trade-off only impacts the redundancy of the system, the improvements are uncontestable.

A. Existing algorithms

There are many proposed clustering schemes in the literature suitable for dynamic wireless sensor networks. Most of them have the same objective, and that is to optimize the resource usage of the network. Nevertheless, for achieving this goal, different criteria are used. Dominant-Set-based [7] protocols aim to reduce the routing cost by finding a Dominant Set in the network. Other clustering schemes [7] try to provide stable cluster architectures so that re-clustering situations are avoided. In this way, the maintenance cost of the network is minimized. Mobility-aware clustering [8] tries to group nodes by their dynamics as movement is usually the main cause for changes in the topology. Other used algorithms try to maximize the life time of mobile devices in a network or to balance the energy consumption amongst all the nodes. Added to these, combined metrics can also be employed to attain a desired clustering scheme.

When dealing with a swarm of satellites, it is difficult to find a clustering algorithm that matches all the requirements: power efficiency, mobility and high data rates. However, there are a few algorithms that are partially fit for the OLFAR project, and, out of these, worth mentioning are Ryu's algorithm for energy-efficient clustering, the global k-means algorithm, the Algorithm for Cluster Establishment (ACE) for uniform cluster formation, the so-called ASH algorithm for highly dynamic networks, or MOBIlity-aware Clustering (MOBIC) for networks that have group mobility behavior.

In his paper [9] Ryu proposes two distributed heuristic clustering schemes that minimize the required transmission energy in two-tiered MANETs. It assumes that the network has only two types of nodes: masters (cluster heads) and slaves (members). A slave node can be connected only to one master, and links between slaves are not allowed. Master nodes are selected in advance, and each master node establishes a cluster based on its connection to the slaves. The clustering process starts with a paging phase in which every master pages the slave nodes with the maximum allowed power. A slave that receives this messages replies with an acknowledgement to the master corresponding to the strongest signal. First, the nodes that receive only one paging signal are allocated with communication channels, and, afterwards, other slaves are allocated channels in the decreasing order of their received power level. By giving priority to slaves that receive only one paging signal and employing power control, this scheme can achieve nearly optimum performance. However Ryu's algorithm uses pre-defined masters and has no method for mobility scenarios.

The global k-means algorithm [10] describes a deterministic global optimization method that minimizes the clustering error (sum of the squared distances between nodes and cluster centers). The algorithm is a fast iterative one that solves the clustering problem with M clusters by solving all intermediate problems with $1, 2, \ldots, M-1$ clusters. The basic idea is that an optimal solution for the M clusters problem can be obtained using a series of local searches with the k-means algorithm. At each local search, $M-1$ cluster centers have their initial optimal position according to the $M-1$ clusters problem. In [10] the method is proposed for a pattern recognition scenario but can be extended for the networking case.

Both ACE [11] and ASH [12] are emergent cluster formation algorithms. In their approaches there is no use of central control or visibility over all the network. Added to this, all the nodes communicate with only a limited number of immediate neighbors. In ACE, the objective to create highly uniform clusters is achieved using two processes. The first controls the spawning of new clusters by having nodes elected as leaders, and the second controls how clusters migrate dynamically to reduce overlap. For instance, a node can decide by itself to become a cluster head. It will broadcast a "Recruit" message to its neighbors that will become the followers of the new cluster. Migration of a cluster is controlled by the leader. Each cluster head will poll its followers to determine the best candidate for the leader of the cluster. Once the best candidate is determined, it will be promoted as the new leader. Thus, the position of the cluster will appear to migrate in the direction of the new cluster head as some of the former followers of the old cluster-head will be no longer part of the cluster, while some new nodes near the new cluster head will become new followers of the cluster [11].

Based also on emergent behavior, ASH [12] tackles the problem of node grouping in networks that exhibit high mobility. Node movement usually introduces a lot of problems in a wireless network, such as routing failure, information loss, and others. The mechanism described in [12] handles mobility in large-scale networks by employing a diffusion process that tends to equalize the pressure in the created groups. By using only local interactions, it creates domains whose centers of gravity move around slowly, providing a quasi-static overlay.

The MOBIC clustering scheme [8] takes mobility into consideration for cluster formation, and, especially, for leader election. It uses the premise that cluster head election is a local process and should only be determined by the neighbors and itself. The algorithm calculates the variance of a node's speed relative to its neighbors, and increases its probability of becoming a master based on that. This is suitable for MANETs in which groups of nodes tend to move with similar speeds and directions. However, if the network is characterized by random movement, MOBIC might not show good performance.

All the algorithms described above have good results in terms of efficiency for different scenarios. Yet, applying them on a complex system such as the OLFAR satellite swarm, will, most probably, cause the system to overload and fail. For instance, Ryu's algorithm and the global k-means algorithm both optimize the energy consumption for a static network. The presence of mobile terminals will cause the cluster structure to be re-built when events take place, and, thus, the performance will be degraded. Mobility-aware schemes generate either large numbers of clusters or multi-hop topologies. For a high-speed network, it is best to avoid these scenarios as much as possible, because of their need for data aggregation.

The algorithm that we propose combines the aforementioned advantages, fitting to the necessities of a satellite swarm. It starts with electing the cluster heads and creating their corresponding clusters, depending on the distribution of nodes. Afterwards, it uses two procedures for node migration and for leader election that keep the system stable and minimize the risk of re-clustering. All the decision-making is done using power metrics, so that, in the permanent regime, the network tends to evolve to minimum power consumption.

B. Assumptions

The following assumptions were made when designing the clustering scheme:

1) The clustering process starts after a gossiping round takes place in the swarm. Hence, every node is aware of the spatial distribution of the swarm.
2) The initialization is done very fast compared to the mobility of the nodes. For the initial cluster formation, we assume that the network is static.
3) All the nodes are either masters or slaves.
4) Every cluster has a star topology. Master nodes are connected in a full-mesh. A slave node can only be connected to one master.
5) A slave node can migrate to another cluster. A master node cannot migrate, unless it changes its role to a slave.
6) For the transmission power, we use the same assumption as in Section III.

$$E_{\text{TX}} = \left(\frac{d_{ij}}{d_{\min}}\right)^2 E \tag{4}$$

where d_{ij} is the distance between sensor i and sensor j.

7) The communication between every pair of nodes is error-free.
8) Receiving and processing power is ignored. After simulating the scenario described in Section III, we concluded that receiving and processing power is a negligible quantity when comparing it to the transmission power.
9) Each master can have, by default, up to N_{ch} slaves. If the number of slaves is larger than N_{ch}, the master will suffer a power penalty. This will be detailed in the following subsections.

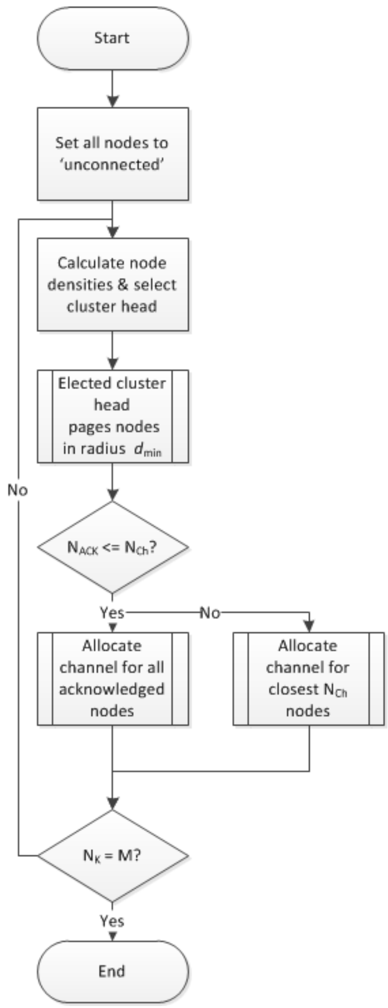

Fig. 5. Initial cluster formation. (N_{ACK} is the number of acknowledgment messages that an elected cluster head receives after paging, N_{K} is the number of formed clusters, and M is the desired number of clusters)

C. Initial cluster formation

In order to choose the nodes that are most suited for the leader role, the influence that a node has on all the other nodes of the network is quantized into a density parameter which is defined below. Using this value, the clusters are formed as described in the flowchart in Fig. 5.

The steps of the cluster formation algorithm are as follows:

1) Calculate node densities.
2) The node with the highest density is elected as the cluster head.
3) The cluster head pages all its neighbors in a radius $d_{\min}$. Nodes that do not belong to any cluster respond with an acknowledgement signal and join the newly formed cluster. If the number of acknowledgements is larger than the number of channels N_{ch}, the cluster head chooses only the closest N_{ch} nodes as his followers.

4) The process is repeated until an apriori chosen number M of clusters is created or all the nodes are connected.

The density of a node i is a power-related parameter and is calculated as:

$$\rho_i = \sum_{\substack{j=1 \\ j \neq i}}^{N} E_{ij}^{-1} \,, \tag{5}$$

where node i is an unconnected node, and E_{ij} is the energy required for transmitting one unit of data from node i to node j, if node j is unconnected, or 0 otherwise.

D. Slave migration

In order to maintain the cluster structure and to keep the power consumption to a minimum, we define a slave migration procedure. The migration mechanism will be controlled entirely by the cluster heads and will work as follows:

1) Each master node will calculate a cost parameter C_{ij} for all the slaves in the network, where C_{ij} approximates the necessary power for node i to be part of cluster j.
2) Nodes that do not belong to any cluster are attributed a default cost value.
3) If for a slave node i, member of cluster j, the following condition is true:

$$\frac{C_{ik}}{C_{ij}} \leqslant \theta_{\text{mig}} \tag{6}$$

then node i cancels its membership to cluster j, and joins cluster k. θ_{mig} is a migration parameter and is equal to 1 in the ideal case. In order to avoid node migrating back and forth between two clusters, as a result of their random movement, the migration constant can be set lower than 1. This way, a node joins a different cluster only if it finds a much better one in terms of cost.

As already mentioned, the cost function of a slave will approximate the necessary power for a slave node to be part of a certain cluster. Initially, we defined the cost C_{ij} as being the necessary power for a node i to communicate with another hypothetical node positioned in the center of gravity of cluster j.

$$C_{ij} = \begin{cases} 1 & , \tilde{d}_{ij} \leqslant d_{\min} \\ (\tilde{d}_{ij}/d_{\min})^2 & , \tilde{d}_{ij} > d_{\min} \end{cases} \tag{7}$$

where $\tilde{d}_{ij}$ is the distance between node i and the center of gravity of cluster j. The coordinates for the centers of gravity are calculated according to (2).

Calculating the cost relative to the center of gravity instead of relating to the position of the master node makes the system more stable. The movement or re-election of the cluster head has less influence on the behavior of the slave nodes. The migration process will be controlled by the entire cluster.

The cost function described by (7) is valid only if the number of slaves in a cluster is less than the number of channels a master can allocate. In most of the clustering algorithms, it is not allowed to have more slaves than the number of channels. Yet, this cannot be the case for our algorithm. We cannot afford to lose data from any satellite.

According to the Shannon-Hartley theorem:

$$C = B \log_2 (1 + SNR) \,, \tag{8}$$

where C is the communication channel's maximum capacity, B is the bandwidth, and SNR is the signal-to-noise ratio.

Taking into account that the communication between every two nodes is ideal, we can proceed as follows. Let us assume that C_n is the channel capacity for accommodating n users, requiring SNR_n. And let us assume that in order to satisfy the requirements of an additional m users a channel capacity C_{n+m} is needed, respectively, SNR_{n+m}. In both of the cases we assume that the available bandwidth B is the same, and also the noise power is the same.

Due to the fact that all users are supposed to have the same requirements, we also have:

$$\frac{C_n}{C_{n+m}} = \frac{n}{n+m} \tag{9}$$

Using (8) and (9) we can deduce a relation between SNR_n and SNR_{n+m}, which describes, in fact, the amount of additional power needed for a cluster to host $n+m$ slaves when it is designated for n slaves.

$$SNR_{n+m} = (1 + SNR_n)^{(1+m/n)} - 1 \tag{10}$$

As a result, in case a cluster will have more than N_{ch} nodes, the necessary power will increase exponentially.

Based on (10), using a Taylor series approximation, and assuming $SNR_n \gg 1$ and m considerably smaller than n, we define a new cost function:

$$C^*_{ij} = \theta_{\text{cost}}(N_j - N_{\text{ch}})C_{ij} \tag{11}$$

for when the number of slaves in a cluster is larger than the number of allocated channels. In (11) θ_{cost} is a parameter depending on SNR_n, N_j is the number of nodes in cluster j (including the potential node i), and C_{ij} is the cost calculated according to (7).

E. Cluster head re-election

Similar to the slave migration process, we define a mechanism for changing the leader of a certain cluster. As the nodes move, a master can turn up to be inefficient as a cluster head, so that another member of the cluster should take its role. The master node should always be the node closest to the center of the group, moving in similar direction and at a similar speed as its members.

The re-election process is a local activity, in which all the members of a certain domain are involved. Let there be a cluster j with master node i. Let G be the center of gravity of the cluster with its position vector $\mathbf{r}_{\text{G}}$ and its speed vector $\mathbf{v}_{\text{G}}$, calculated according to (2), respectively:

$$\mathbf{v}_{\text{G}} = \bar{\mathbf{v}}_i \tag{12}$$

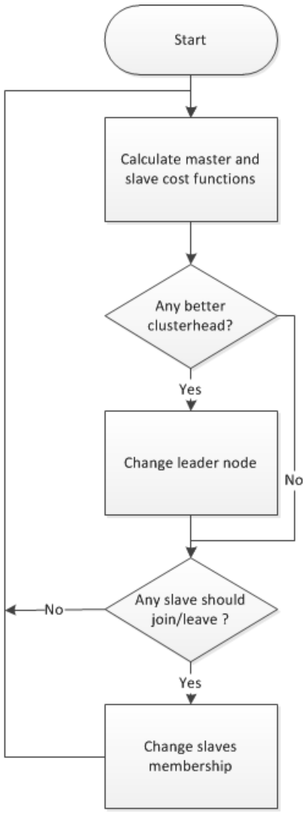

Fig. 6. Master Node Task.

where $\mathbf{v}_i$ are the speed vectors of nodes i in the cluster j.

With these definitions, the decision to attribute the role of the leader to a new node is made based on the following condition:

$$(1-\alpha)\frac{\|\mathbf{r}_i-\mathbf{r}_{\mathrm{G}}\|}{\|\mathbf{r}_j-\mathbf{r}_{\mathrm{G}}\|}+\alpha\frac{\|\mathbf{v}_i-\mathbf{v}_{\mathrm{G}}\|}{\|\mathbf{v}_j-\mathbf{v}_{\mathrm{G}}\|}\leqslant\theta_{\mathrm{CH}}\,, \qquad (13)$$

where i denotes the slave node that candidates for a leader position, and j is the actual master node. α is a tunable parameter for strengthening one of the two terms, and θ_{CH} is a parameter that has the same role as θ_{mig} defined in the previous section.

In Fig. 6 a flowchart of the entire process of a master node is shown.

VI. Simulations

We simulated the algorithm using a Netlogo environment and Matlab. The test scenario consisted of a network of 100 nodes uniformly distributed on square plane. The number of clusters was selected to be 6, and for every cluster 16 channels were allocated without any penalty. For all the nodes, a minimum transmission range of $L/4$ was chosen. The default cost value for unconnected nodes was set to four. According to this value, it is possible for unconnected terminals to join a cluster only when the distance to the center of gravity is less than $2d_{\min}$.

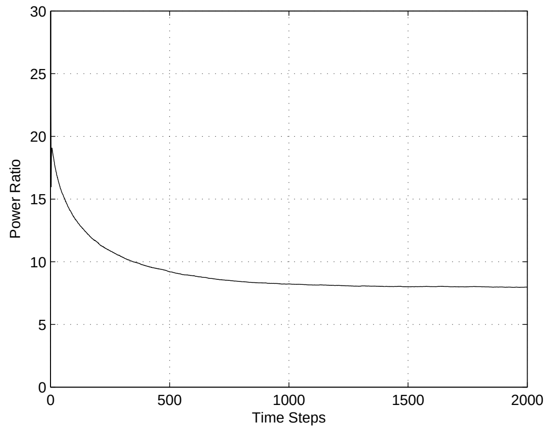

Fig. 7. Power Ratio: transmission power needed for a full-mesh topology divided by the transmission power necessary when applying the dynamic clustering scheme.

The movement pattern was chosen to be a random walk pattern: each node was given a speed of $L/200$ per time step and could change its movement direction at every simulation step with a probability of 0.1. Finally, we set the parameters θ_{cost}, θ_{mig}, θ_{CH} and α with the values 2, 0.6, 0.6 and 0, the last one corresponding to a random movement scenario.

The results of the numerical simulations are statistic results, which are the mean values of 1,000 random configurations of the network. As performance metrics, we use the total consumed power for transmission and the average cluster size. We ignored the receiving and processing consumed energy as it is very low comparing it to the transmitting energy [13]. The total consumed power consists of the power necessary for all the existing links in the considered network: slave-master, master-slave, and master-master links. We compare this value with the total necessary power in case a full-mesh structure would be used. In Fig. 7, we plotted the ratio between the two values, i.e. the power needed for a full-mesh topology divided by the power needed when we apply our clustering scheme. After the initial cluster formation, the value of the ratio will be very high because some of the nodes will be unconnected. In time, the coverage of the clusters grows to 100% and the value tends to stabilize. As anticipated in Section IV, the clustered network is more power-efficient.

The average number of nodes in a cluster and the standard deviation are plotted in Fig. 8. The proposed clustering scheme generates quasi-equal-sized domains in term of number of nodes. The mean value will remain constant once every node is assigned to a cluster. However, the standard deviation indicates that cluster sizes will vary slightly, in order to keep the cost functions to a minimum.

The proposed algorithm comes with a few drawbacks, some of which are shown in Fig. 9 and Fig. 10. Node mobility will generate node transitions from one cluster to the other, and role-transitions from master to slave and vice-versa. Changes in the node distribution will not generate re-clustering, but

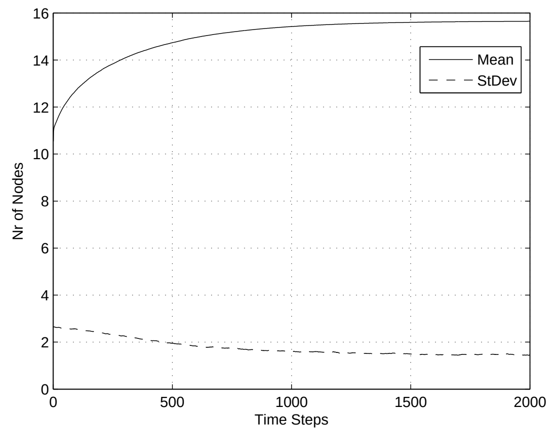

Fig. 8. Cluster Size: average number of nodes per cluster and standard deviation

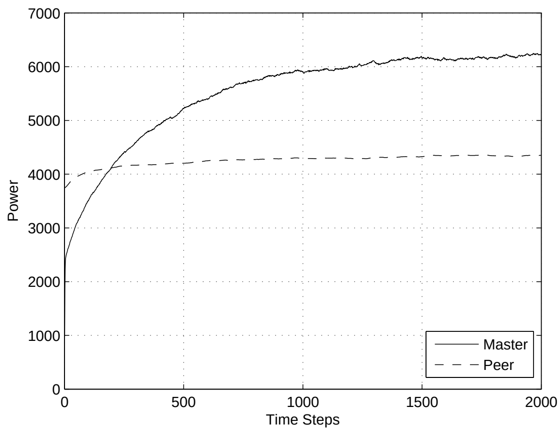

Fig. 10. Power Requirements: maximum transmission power for master nodes (clustering approach) and peer nodes (full-mesh topology).

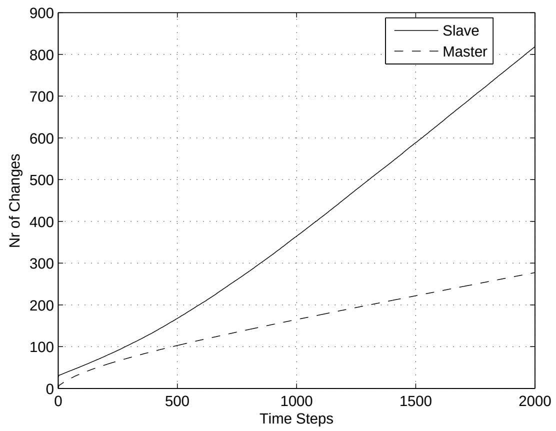

Fig. 9. Cumulative number of transitions.

will have a major impact in the roles of the swarm members. Initially, the cluster structure will not be very stable, as a large number of nodes will tend to join the newly formed clusters. This results in a high number of master-slave and slave-master transitions (dashed line). After a transient phase, cluster head re-election process will occur less often, yet the number of slave migrations will increase. Added to this, as anticipated, the power requirements for master nodes will prevent them from having other tasks than communication.The main drawback of the algorithm is common for most of the clustering schemes. The power requirements for the master node will prevent cluster heads from carrying any other tasks.

VII. Conclusions

In this paper, we presented an adaptive topology that matches the demands of the OLFAR satellite swarm. By employing power cost functions and a node migration mechanism, the proposed algorithm creates a two-layer hierarchical structure that improves the data distribution in the system. Power consumption for communication will mainly be concentrated in only a few satellites, leaving the other members of the swarm with enough resources to fulfill the low-frequency observation task.

Although we designed the presented mechanisms for a particular application, they can be used for most WSNs with similar characteristics. By simply tuning the parameters, the scheme may be a solution for many distributed systems that exhibit mobility.

References

[1] M. J. Bentum, C. J. M. Verhoeven, A. J. Boonstra, A. J. van der Veen and E. K. A. Gill, "A novel astronomical application for formation flying small satellites," *60th International Astronautical Congress*, Daejeon, Republic of Korea, 12–16 October 2009.

[2] R. T. Rajan, S. Engelen, M. J. Bentum and C. J. M. Verhoeven, "The orbiting low frequency antenna array," *IEEE Aerospace Conference*, Big Sky, MT, 5–12 March 2011.

[3] S. Engelen, E. K. A. Gill and C. J. M. Verhoeven, "Systems Engineering Challenges for Satellite Swarms," *IEEE Aerospace Conference*, Big Sky, MT, 5–12 March 2011.

[4] S. Jester and H. Falcke, "Science with a lunar low-frequency array: From the dark ages of the universe to nearby exoplanets," *New Astronomy Reviews*, 2009.

[5] N. Saks, A. J. Boonstra, R. J. Rajan, M. J. Bentum, and F. Belien, "DARIS, a fleet of passive formation flying small satellites for low frequency radio astronomy," *Small Satellite and Services Symposium*, Funchal, Portugal, 31 May–4 June 2010.

[6] D. Kempe, J. Kleinberg and A. Demers, "Spatial gossip and resource location protocols," *Journal of the ACM*, vol. 51, no. 6, November 2004.

[7] J. Yu and P. Chong, "A survey of clustering schemes for mobile ad hoc networks," *Communications Surveys Tutorials, IEEE*, vol. 7, no. 1, pp. 32–48, 2005.

[8] P. Basu, N. Khan, and T. D. C. Little, "A Mobility Based Metric for Clustering in Mobile Ad Hoc Networks," *Proc. IEEE ICDCSW01*, pp. 413–18, April 2001.

[9] J. H. Ryu, S. Song and D. H. Cho, "New Clustering Schemes for Energy Conservation in Two-Tiered Mobile Ad-Hoc Networks," *Proc. IEEE ICC01*, vo1. 3, pp. 86266, June 2001.

[10] A. Likas, N. Vlassis and J. J. Verbeek, "The global k-means clustering algorithm," *Pattern Recognition*, vol. 36, pp. 451-461, 2003.

[11] H. Chan and A. Perrig, "ACE: An emergent algorithm for highly uniform cluster formation," *European Conference on Wireless Sensor Networks*,Berlin, Germany, 19–21 January, 2004.

[12] A. Pruteanu, S. Dulman and K. Langendoen, "ASH: Tackling node mobility in large-scale networks," *IEEE International Conference on Self-Adaptive and Self-Organizing Systems*, Budapest, Hungary, September 27–October 1 2010.
[13] G. J. Pottie and W. J. Kaiser, "Wireless Integrated Network Sensors," *Communications of the ACM*, vol. 43, no. 5, pp 51-58, May 2000.